

Towards modeling of epidemic spread: eigenvalue computation

Z.Liu^{a,c}, N.Emad^{a,c}, S.B.Amor^a, M.Lamure^b

^a*University of Versailles, 45, avenue des Etats-Unis, 78035 Versailles, FRANCE*

^b*University of Lyon 1, 5, avenue Pierre Mendès France, 69676 Bron, FRANCE*

^c*Maison de la Simulation, bat. 565, CEA Saclay, 91191 Gif-sur-Yvette, FRANCE*

Abstract

The eigenvalue equation intervenes in models of infectious disease propagation and could be used as an ally of vaccination campaigns in the actions carried out by health care organizations. The epidemiological modeling techniques can be considered by analogy, as computer viral propagation which depends only on the underlying graph status at a given time. We point out pagerank as method to study the epidemic spread and consider its calculation in the context of small-world phenomenon. Basing on Barabási-Albert power law graphs, we adapt the model to make the matrix involved have a large and sparse structure, so that numerical methods based on matrix-vector products could be employed. We propose to use the implicitly restarted Arnoldi method and introduce its adapted parallel algorithm for the case of very large social graphs.

Keywords: Epidemic, pagerank, Complex networks, Non hermitian eigenvalue problem, Sparse matrix

1. Introduction

Dynamic complex systems appear in many areas such as physics, biology, and computer networks etc. In the domain of health research, quick response and effective control of widely spreading health crises stay a big challenge for public health officials as well as scientists. In order to simulate the epidemic

Email addresses: `zifan.liu@prism.uvsq.fr` (Z.Liu), `nahid.emad@prism.uvsq.fr` (N.Emad), `soufian.benamor@prism.uvsq.fr` (S.B.Amor), `michel.lamure@univ-lyon1.fr` (M.Lamure)

spreading, such as H1N1 outbreak in France, traditional models need hundreds of experiments and compute the expected outcome by averaging. In addition, these experiments should be adjusted on a daily basis during the initial outbreak. For example, Network Dynamics and Simulation Science Laboratory (NDSSL) has proposed a parallel simulation model Simdemics [1, 2], designed to scale to the entire United states (300 Million people). This solver generates roughly 300 GB of data and is expected to increase as more details is added. One run takes 3000 cpu hours on a 1.5TF machine of 448 cores and one experiment takes 100 to 300 runs. As a result, They could only expect 1 to 4 experiments per year. That's not fast enough to answer urgent requests during the beginning phase of outbreak.

There are two deterministic processes being used in the analysis of epidemic spreading, the susceptible-infective-recovered (SIR) process and the susceptible-infective-susceptible (SIS) process, where the difference is whether or not the disease confers lifelong immunity. Most of the existing work in this area has focused on considering the effect of underlying network structure on epidemic dynamics by using tools from statistical physics [3]. The new trend of complex network based models recognize the individual-level randomness and network topology as significant factors on the dynamic of epidemics, which introduces stochastic aspects in the modeling [4, 5]. Adaptive network models can be seen as an offspring of this tendency, the idea of which is that individuals usually attempt to reduce their risk of infection by eliminating contact with contagious individuals [6, 7, 8]. In other words, the particularity of this approach is to model infection spreading in a population with evolving contacts. However, the above models all need heavy simulations when facing with a large population because more details about the population should be used. Consequently, it is difficult for them to establish effective vaccination campaigns during the beginning phase of an urgent outbreak. We argue that these difficulties could be well handled within an eigenvalue model and discuss its computation using technology of high performance computing.

Before introducing the proposed model, a brief review of epidemiological models is given in the next section. The proposed pagerank-like model is presented in section 3. Section 4 gives the reason why we choose IRAM as computation method to solve the underlying eigenproblem. Parallel resolution of this eigenproblem is discussed in section 5. Section 6 is devoted to numerical simulation.

2. Review of epidemiological models

One common used measure of infectivity is the epidemic threshold λ_c , which is the minimum infectiousness that a disease has to have in order to invade a network. Each susceptible node is infected with rate ν if it is connected to one or more infected nodes. At the same time, infected nodes are cured with rate δ , defining an effective spreading rate $\lambda = \nu/\delta$. If the value of λ is above the threshold, $\lambda \geq \lambda_c$, the infection spreads and becomes persistent. Below it, $\lambda < \lambda_c$, the infection dies out exponentially fast [9, 10].

The homogeneous models (SIR or SIS) assume that the population mixed at random, so that each individual has an equal chance of coming into contact with any other individual [11]. However in real world, it is not rare to find the different mixing rates between the population subgroups [12]. So a direct improvement is to avoid the random-mixing assumption. Models that include underlying network structure achieve this goal by assigning to each individual a finite set of contacts [13, 14].

The difference between various network models depends on how individuals are distributed in space and how connections are formed. The Paul Erdős and Alfréd Rényi model (E-R model) consists of n nodes, joined by edges which are placed between pairs of nodes chosen with equal probability p . By using ideas drawn from percolation theory [15], it is found that there is a non-zero epidemic threshold,

$$\lambda_c = \frac{1}{\langle k \rangle} \quad (1)$$

where $\langle k \rangle$ is the average connectivity. The degree distribution of this model is Poisson and the epidemic growth rate in such a network is reduced in comparison with the random mixing model [16]. Despite being one of the oldest and best studied models of a network, E-R model differs from real networks in two crucial ways: it lacks network clustering and it has an unrealistic Poissonian degree distribution [17]. By the following, we introduce two other network models to remedy these two inconvenients.

Models based on Small-world networks got into our sight by the work of Watts and Strogatz [18, 19]. Small worlds can be formed by adding a small number of random connections to a lattice¹. And Newman's work

¹Lattices display high clustering but long path lengths because connections are estab-

[20] gives another effective way of constructing this kind of networks. Random networks display low clustering but short path lengths since there are many long-range links, whereas small-world networks have high clustering and short path lengths. These characteristics have important implications in the context of epidemics: the high level of clustering means that most infection occurs locally, but short path lengths means that epidemic spreading through the network is rapid [18]. By applying the percolation theory to small-world networks to calculate the threshold, it is found that [21]:

$$\lambda_c = \frac{\sqrt{1 + 12\phi + 4\phi^2} - 1 - 2\phi}{4\phi} = 1 - 4\phi + O(\phi^2) \quad (2)$$

where, it is assumed that each individual is linked to its two nearest neighbors and on average to ϕ randomly chosen other individuals.

In 1998, a project to map the World Wide Web has revealed a surprising fact that a few highly connected pages are essentially holding the World Wide Web together. Counting how many Web pages have exactly k links showed that the degree distribution followed a power-law. Following researches observed many real world networks that display this phenomenon, while small worlds, random networks have a power-law degree distribution. Scale-free networks can be constructed dynamically by adding new individuals to a network one by one with preferential mechanism. The major contribution of this model is the heterogeneity in numbers of contacts and the existence of hubs (the most highly connected nodes in the network) [22]. Hubs in a network play a pivotal role in the spread and maintenance of infection. Research suggests that the simultaneous elimination of as few as 5 to 15 percent of all hubs can crash a system. Despite some practical difficulties, immunizations targeting hubs could be interesting [23, 24, 25]. Further research indicated the absence of epidemic threshold in scale-free networks [9]. That is, even weakly contagious viruses will spread and persist in the system.

To summarize this section, we focus on the assumptions used for each model and on the existence of epidemic threshold. Recently, an interesting study has proved the close relationship between the epidemic threshold of a network and the largest eigenvalue of network's adjacency matrix, which can subsume many previous known threshold for special case graphs (E-R, BA

lished between adjacent individuals in a 2-D grid.

power-law, homogeneous) [26].

3. The proposed model

We propose to make use of Google's pagerank model [27] by analogy. An individual in a social graph is analogous to a webpage in a web graph. The common concept between pagerank model and our epidemic model is the random walk. In pagerank model, the surfer (or walker) starts from a random page, and then selects one of the outlinks from the page in a random fashion. Each page has two states as being visited by surfer or not. The pagerank (importance) of a specific page represents the probability that the surfer is present at this page. In our epidemic model, the virus could be viewed as a walker and its propagation could be viewed as a path that consists of a succession of random steps. Each individual has two states as being infected or not. The pagerank (importance) of a specific individual represents the probability that the virus reaches this individual during the course of epidemic. To use mathematical formalism, let $G = (V, E)$ be a directed graph with individuals set V and outlinks set E . The graph might be directed. That means, if there is a link $i \rightarrow j$ in graph G where $i, j \in V$, $j \rightarrow i$ is not necessarily true. For example, blood disease could only happen from the donators to the acceptors. Suppose that the graph G has n individuals with degree $d = (d_1, d_2, \dots, d_n)$, where d_j is the number of links individual j has to other individuals. At each time step t , the virus has a state $s_t \in V$, indicating which individual it is on at time t . If $s_t = i$, then at time step $t + 1$, the virus moves to one of his neighbors j chosen uniformly at random from all of i 's neighbors.

$$P_{j,i} = P[s_{t+1} = j | s_t = i] = \begin{cases} \frac{1}{d_i} & \text{if } i \rightarrow j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The probability $P(s_t = i)$ for $i \in V$ depends only on s_{t-1} and not on s_{t-2}, s_{t-3}, \dots , so that $\{s_t\}$ is a Markov chain. $\{s_t\}$ is characterized by its initial state and a transition matrix P , given by $P_{j,i} = P(s_t = j | s_{t-1} = i)$ with $P_{j,i} \in [0, 1]$ for all $i, j \in V$ and $\sum_{j \in V} P_{j,i} = 1$. According to Frobenius theorem, $\lambda = 1$ is one of the eigenvalue of the transition matrix P and is the biggest eigenvalue. Thus, there is a stationary distribution for the final state of epidemic spread. Let x_i be the probability that individual i is infected during epidemic and we write the stationary distribution as $x = (x_1, x_2, \dots, x_n)$ for

the whole population. This infection vector x is independent of the starting distribution and has the relationship: $Px = 1 * x = x$. To sum up, the infection vector x 's implication in social graph is similar to that of pagerank vector in web graph. The problem consists, as a result, to find the dominant eigenvector with 1 as eigenvalue for the transition matrix P of the social graph. An example of 5 individuals has been given in Figure 1. The set

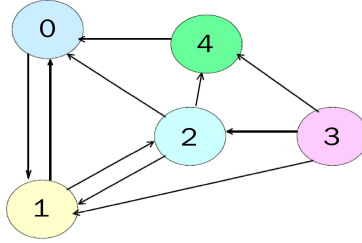


Figure 1: Small social graph of 5 individuals

$V = \{0, 1, 2, 3, 4\}$ and the set $E = \{0 \rightarrow 1, 1 \rightarrow 0, 1 \rightarrow 2, 2 \rightarrow 0, 2 \rightarrow 1, 2 \rightarrow 4, 3 \rightarrow 1, 3 \rightarrow 2, 3 \rightarrow 4, 4 \rightarrow 0\}$. In matrix formulation, these links give the transition matrix P as:

$$P = \begin{pmatrix} 0 & 1/2 & 1/3 & 0 & 1 \\ 1 & 0 & 1/3 & 1/3 & 0 \\ 0 & 1/2 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 0 \end{pmatrix} \quad (4)$$

The first difficulty with this model is the existence of dangling individuals [28], which containing no outlinks. These individuals will result in one or more columns of zeros in transition matrix P . For our example in Figure 1, if we delete the link $0 \rightarrow 1$, then the first column of matrix P will contain only zeros. Several ideas have been proposed to deal with this problem [29, 30]. Research by the initial pagerank paper [27] indicates that the pagerank could be calculated by removing the links to dangling pages from the web graph. However, theoretically this process might generate new dangling pages and iteratively remove all pages from the graph. In the context of epidemic spread, dangling individuals could be considered as deadends for virus' random walk process. So a nature thinking is to add a loop with probability

1 to these persons themselves. By this way, diagonal elements corresponding to dangling individuals in matrix P are filled with 1. We adopt this simple solution.

The second difficulty with this model is the problem of non-unique rankings. The phenomenon of “small-world” reveals the clustering effect in social networks. Since very few links exist between clusters, some isolated clusters will break the strong connectivity of graph. It can be shown that the transition matrix P will not yield a unique ranking vector x with such isolated clusters [28]. The common solution is to add a jumping vector to the random walk process:

$$A = \alpha P + (1 - \alpha)ve^T \quad (5)$$

where A is disease transition matrix, v is the teleportation vector, e is the vector $[1, \dots, 1]^T$ and α , the damping factor, is a positive parameter smaller than 1. The virus has a small probability $(1 - \alpha)$ to jump from any individual to any other individual in a social graph. This would happen, for example, when an infected person (virus carrier) meets someone outside his normal contacts. Considering the preferential attachment of social scale-free networks [12], we choose v to be proportional to individuals’ degree and normalizes it by “1-norm”. In short, there is a small probability that an individual establishes a new temporary link with someone who already has many links.

$$v = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix}, v_i = \frac{d_i}{\sum_{i=1}^n d_i} \quad (6)$$

All column sums of the new transition matrix A are 1, so A is still a stochastic matrix with dominant eigenvalue equal to 1: $Ax = x$. In the example of

Figure 1, $d = (1, 2, 3, 3, 1)$, $\sum_{i=0}^{i=4} d_i = 10$. By taking $\alpha = 0.9$, we have:

$$\begin{aligned}
A &= \alpha P + (1 - \alpha)ve^T \\
&= 0.9 * \begin{pmatrix} 0 & 1/2 & 1/3 & 0 & 1 \\ 1 & 0 & 1/3 & 1/3 & 0 \\ 0 & 1/2 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 0 \end{pmatrix} + 0.1 * \begin{pmatrix} 1/10 \\ 2/10 \\ 3/10 \\ 3/10 \\ 1/10 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}^T \\
&= \begin{pmatrix} 0.01 & 0.46 & 0.31 & 0.01 & 0.91 \\ 0.92 & 0.02 & 0.32 & 0.32 & 0.02 \\ 0.03 & 0.48 & 0.03 & 0.33 & 0.03 \\ 0.03 & 0.03 & 0.03 & 0.03 & 0.03 \\ 0.01 & 0.01 & 0.31 & 0.31 & 0.01 \end{pmatrix}
\end{aligned} \tag{7}$$

It must be noticed that our proposed model is based on SIS epidemiological process where, at a given time, each individual can be susceptible (S) or infected (I). So we suppose that individuals recover with no immunity to the disease, that is, individuals are immediately susceptible once they have recovered. Furthermore, the characteristic of individuals as well as that of virus are not taken into account. The model depends only on the status of social graphs at a given time. Infection vector x could help health officials to decide the relative importance of different agents in a population facing an epidemic. This is especially useful when the resource of vaccination are limited during the beginning phase of an urgent outbreak. Priorities should be given to those individuals with bigger ranking in x . In addition, a fast computation of this vector could be expected thanks to the efficient implementation of eigenvalue algorithm.

4. Computation methods

An efficient solution to a large sparse eigenvalue problem strongly depends on the proper choice of iterative methods. Our first objective is to choose the best method to calculate the dominant eigenvector.

A lot of research found that the damping factor α strongly affects the convergence of iterative methods [31, 32]. So another special attention has been paid to investigate how the convergence of the proposed algorithm is influenced by this degree of teleportation.

Algorithms based on matrix-vector product (MVP) might be advantageous. Suppose x is a vector of p -norm 1, $Ax = \alpha Px + (1 - \alpha)v(e^T x)$ where $e^T x$ is a scalar. So the MVP of A is expressed as MVP of a sparse matrix P plus a vector. Otherwise, direct computation of Ax for a dense matrix A (see example in (7)) is bottlenecked by memory on large graphs. The simple power method can be used to solve this non-hermitian eigenvalue problem. The idea is to write the initial vector x_0 as a linear combination of $\sum_{j=1}^n \alpha_j v_j$, where v_j are eigenvectors of A . Without loss of generality, suppose λ_1 is the dominant eigenvalue, we have:

$$\begin{aligned}
x_k &= Ax_{k-1} = A^2 x_{k-2} = \dots = A^k x_0 \\
&= A^k \sum_{j=1}^n \alpha_j v_j = \sum_{j=1}^n \alpha_j A^k v_j = \sum_{j=1}^n \lambda_j^k \alpha_j v_j \\
&= \lambda_1^k (\alpha_1 v_1 + \sum_{j=2}^n (\lambda_j / \lambda_1)^k \alpha_j v_j)
\end{aligned} \tag{8}$$

For $j > 1$, $|\lambda_j / \lambda_1| < 1$, so that $(\lambda_j / \lambda_1)^k \rightarrow 0$, leaving only the pagerank eigenvector v_1 . Employing the known eigenvalue 1 as a shift offers potential for accelerating convergence. Besides, based on the fact that the smallest eigenvalue of A is the reciprocal of the largest eigenvalue of A^{-1} , shifted inversed version of power method is described in Algorithm 1. The convergence rate of this algorithm is $|(\lambda_j - \mu) / (\lambda_k - \mu)|$, where λ_j and λ_k are eigenvalues of A such that $|\lambda_j - \mu|^{-1}$ is the largest and $|\lambda_k - \mu|^{-1}$ is the second largest eigenvalue of $(A - \mu I)^{-1}$. On one hand, the closer the damping factor is to 1, the closer the matrix A is to the original transition matrix P . If the largest eigenvalue is not well separated from other eigenvalues, then this rate will be close to 1 and the convergence of the method will be very slow. On the other hand, inverse iteration requires to solve a linear system for the large matrix $(A - \mu I)$, adding both computational and storage complexity.

To remedy the previous difficulties, Krylov subspace method can be an alternative. Krylov subspace method allows approximation of an eigenpair (λ, x) of A by a Ritz-elements pair $(\mu \in \mathbb{C}, x^{(m)} \in \mathbb{K}_m)$ where the subspace \mathbb{K}_m is defined by

$$\mathbb{K}_m = \text{Span}\{x_0, Ax_0, \dots, A^{m-1}x_0\} \tag{9}$$

As shown in power method, the vectors $A^j x$ point more and more in the direction of the dominant eigenvector as j increases. This means that for

<p>Input: $A(n \times n)$: the disease transition matrix with each column sum as 1, x_0: the initial guess of the infection eigenvector, μ: the shift nearest to the desired eigenvalue (here $\mu = 1$). Output: x</p> <pre> 1 $x = x_0 / \ x_0\$; 2 for $j = 1, 2, 3, \dots$ <i>until</i> λ converge do 3 Solve $(A - \mu I)w = x$; 4 $\lambda \leftarrow \mu + \frac{x^* w}{w^* w}$; 5 $i \leftarrow i_{\max}(w)$; 6 $x \leftarrow x / (e_i^T w)$; 7 end 8 return x; </pre>

Algorithm 1: Inverse Iteration Method

large m , most of vectors in Krylov subspace will point in about the same direction. The basis is thus ill-conditioned. Generally, it is desired to build a well-conditioned basis $\{w_1, w_2, \dots, w_m\}$ for \mathbb{K}_m . One popular choice is an orthogonal basis which leads to Arnoldi procedure. This method approximates k eigenpairs of A by those of a matrix of order m , where $k \leq m \ll n$. This matrix, designated by H_m is obtained by orthogonal projection of A onto an m -dimensional subspace \mathbb{K}_m . Let W_m be the matrix whose column w_1, w_2, \dots, w_m constitute an orthogonal basis of \mathbb{K}_m . The problem is to find $\mu \in \mathbb{C}$ and $y \in \mathbb{C}^m$ such that

$$(H_m - \mu I)y = 0 \tag{10}$$

where the matrix H_m of dimension $m \times m$, is defined by $H_m = W_m^* A W_m$. Note that W_m^* is the transpose conjugate of W_m and $x = W_m y$. Therefore, some eigenvalues of A can be approximated by the eigenvalues of the matrix H_m . Solving the problem (10) is relatively easy thanks to the Hessenberg structure of matrix. The basic Arnoldi procedure with the orthogonalization refinement [33] is described in Algorithm 2. The sequence of statements in if-clause assures that the new direction f_{j+1} is numerically orthogonal to the previously computed directions, i.e., to the columns of W_{j+1} . A larger η sets more strict condition for orthogonality [34]. Yet a disturbing aspect of Arnoldi method is that the number of iterations (the size of the subspace

Input:
 $A(n \times n)$: the disease transition matrix with each column sum as 1,
 w_0 : the starting vector,
 m : the size of subspace.

Output:
 w_1, w_2, \dots, w_m : an orthogonal basis of Krylov subspace,
 $H(m \times m)$: the $m \times m$ Hessenberg matrix.

```

1  $w_1 = w_0 / \|w_0\|$ ;
2  $v = Aw_1$ ;
3  $h = w_1^* v$ ;  $f_1 = v - w_1 h$ ;
4  $W_1 \leftarrow [w_1]$ ;  $H_1 \leftarrow [h]$ ;
5 for  $j = 1, 2, 3, \dots, m - 1$  do
6    $\beta_j = \|f_j\|$ ;
7    $w_{j+1} = f_j / \beta_j$ ;
8    $W_{j+1} \leftarrow [W_j, w_{j+1}]$ ;  $\widehat{H}_j \leftarrow \begin{bmatrix} H_j \\ \beta_j e_j^* \end{bmatrix}$ ;
9    $v = Aw_{j+1}$ ;
10   $h = W_{j+1}^* v$ ;
11   $f_{j+1} = v - W_{j+1} h$ ;
12  if  $(\|f_{j+1}\| < \eta \|h\|)$  then
13     $s = W_{j+1}^* f_{j+1}$ ;
14     $f_{j+1} = f_{j+1} - W_{j+1} s$ ;
15     $h = h + s$ ;
16  end
17   $H_{j+1} \leftarrow [\widehat{H}_j, h]$ ;
18 end

```

Algorithm 2: m -Step Arnoldi Procedure

m) needed to compute the desired eigenpair is unknown, except with a very fortunate choice of starting vector w_0 . The basic algorithm increases m until the dominant eigenvalue of A are found. For storage, in addition to A , the method keeps m vectors of length n and an $m \times m$ Hessenberg matrix, which gives $O(nm + m^2/2)$. For computation complexity, matrix-vector product costs about $(2m * nnz)$ operations, where nnz is the number of nonzero elements in A . The modified Gram-Schmidt procedure costs $O(m^2n)$ operations. Since the size n of a social network may attain millions or even billions of individuals, increasing m causes both storage and computational overhead.

One way to avoid this difficulty is by restarting techniques. The idea is to restart the iteration with a vector that has been preconditioned so that it is more nearly in a k -dimensional invariant subspace which contains the dominant eigenvector. As stated before, the most consuming part in Arnoldi procedure is the MVP due to the very large size of the matrix. Implicitly restarted Arnoldi method (IRAM) [35, 36], which uses truncated implicitly shifted QR iterations [37], provides a means to reduce the number of MVP needed by Arnoldi procedure from m to $r = m - k$, with $(m - k)$ the number of shifts used in QR iterations. The sequential algorithm of IRAM is described in Algorithm 3. Each of these shift cycles, in step 6 – 10 of Algorithm 3, results in the implicit application of a polynomial in A of degree r to the starting vector [35].

$$w_0 \leftarrow \psi(A)w_0, \text{ where } \psi(\lambda) = \prod_{j=1}^r (\lambda - \mu_j) \quad (11)$$

The roots of this polynomial are the shifts used in the QR process and these should be selected to filter unwanted information from the starting vector and hence from the Arnoldi factorization. In our case, to find the dominant eigenpair, we could choose as shifts the r eigenvalues with smallest moduli from the spectrum of H_m .

Concerning the stopping criteria of Algorithm 3, define the vector $x = W_m y$ to be a Ritz vector associated with Ritz value μ .

$$\| AW_m y - W_m H_m y \| = \| Ax - \mu x \| = \| f_m \| \| e_m^* y \| \quad (12)$$

By using the backward error associated with IRAM [38], convergence test in step 3 of Algorithm 3 is: $\| f_m \| \| e_m^* y \| < \| A \|_F \epsilon$ where e_m is m^{th} vector

Input:
 $A(n \times n)$: the disease transition matrix with each column sum as 1,
 w_0 : the starting vector,
 m : the size of subspace,
 r : the number of shifts and $m = r + k$.

Output:
 x : the dominant eigenvector associated with eigenvalue 1.

```

1  $w_1 = w_0 / \|w_0\|$ ;
2 compute the  $m$ -step Arnoldi factorization:  $AW_m = W_m H_m + f_m e_m^*$ ;
3 while not converge do
4   compute the spectrum of  $H_m$  ( $\sigma(H_m)$ ) and select  $r$  shifts
      $\mu_1, \mu_2, \dots, \mu_r$ ;
5    $Q = I_m$ ;
6   for  $j = 1, 2, \dots, r$  do
7     QR factorization:  $Q_j R_j = H_m - \mu_j I$ ;
8      $H_m = Q_j^* H_m Q_j$ ;
9      $Q = Q Q_j$ ;
10  end
11   $\beta_k = H_m(k+1, k); \sigma_k = Q(m, k)$ ;
12   $f_k = w_{k+1} \beta_k + f_m \sigma_k$ ;
13   $W_k = W_m Q(:, 1:k); H_k = H_m(1:k, 1:k)$ ;
14  begin with the  $k$  step Arnoldi factorization  $AW_k = W_k H_k + f_k e_k^*$ ,
    apply  $r$  additional steps of the Arnoldi procedure to obtain a new
     $m$ -step Arnoldi factorization  $AW_m = W_m H_m + f_m e_m^*$ 
15 end

```

Algorithm 3: IRAM

of the canonical basis of \mathbb{C}^m and ϵ is the tolerance. Since the dominant eigenvalue 1 is known, when a Ritz value has converged to 1, IRAM iterations could be stopped.

5. Analysis of parallelism

We look for individuals or group of people most likely to spread the disease. We have seen that these individuals or groups are determined according to the network status of individuals at a given time. To take into account the networks of realistic size and to find quick solutions to stop the spread of disease, we must perform effective parallel implementations of the methods.

IRAM consists of four main tasks. First, the projection phase: step 1, 2 in Algorithm 3, manipulates the n -sized data sets for sparse MVP. The second phase which includes implicitly shifted QR iterations, step 4–12 in Algorithm 3, acts on m -sized data sets. The third phase constructing the r additional steps of Arnoldi factorization, step 13 – 14 in Algorithm 3, manipulates on n -sized data sets for sparse MVP as well. At last the convergence test, step 3 in Algorithm 3, deals with n -sized data sets to calculate $\|f_m\|$. Because phase one and three constitute the most expensive part of the algorithm, we propose to distribute them between processors and to run phase two and four redundantly on all processors.

6. Stochastic simulation using the infection vector

In this experiment, we use a small graph *ba* to simulate the real world epidemic spread with distribution of vaccination. We consider people who receive vaccination being permanently immunized against viruses. For larger graph, parallelization will be needed due to the memory requirement but the implementation of such parallel simulator is not the objective of the test.

We assume a universal infection rate ν , a jumping rate α (damping factor) and a curing rate δ for every individual. Before each simulation, we randomly choose a set of infected individuals. Then the propagation of virus proceeds by time step. During each time step, an infected individual infects each of its neighbors with probability ν . And this infected individual also passes the disease to another random chosen non-neighbor by probability α . Additionally, every infected individual is cured with probability δ . The result is the average over 10 runs and it is presented in Figure 2. Here, we compare three cases. First of all, without distribution of vaccination, we try to give

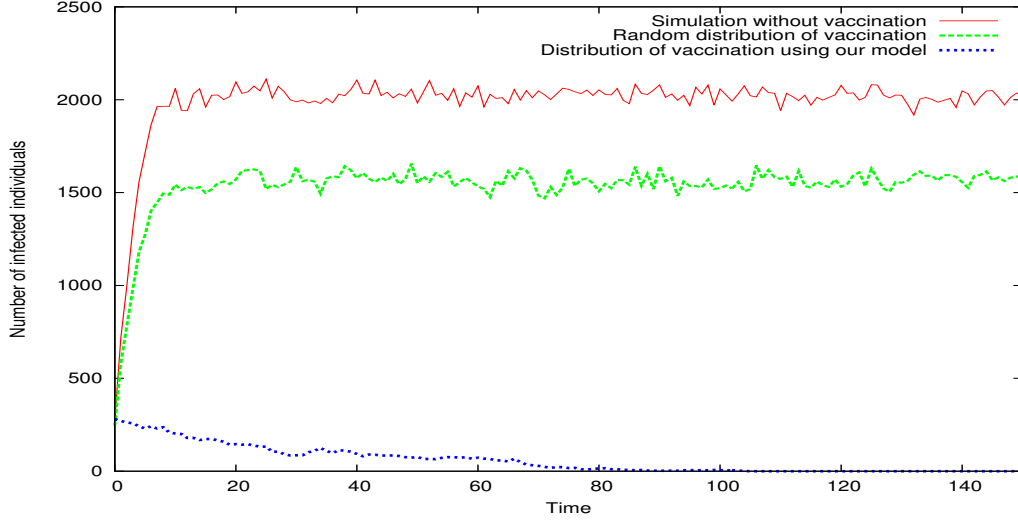


Figure 2: Time series of infection in an 7010-node power-law social graph ba , with $\alpha = 0.85$, $\nu = 0.2$ and $\delta = 0.24$

the worst case for time evolution of infection. Secondly, with random distribution of vaccination, we begin the simulation by distributing vaccination to a random chosen group of individuals. Then, we simulate time evolution of infection. Thirdly, with distribution of vaccination using our pagerank-like model, we calculate the infection vector for the underlying social graph and then distribute vaccination to individuals with big ranking in the vector.

The figure verifies the absence of epidemic threshold in scale-free networks. Without interventions, the epidemic will always enter an endemic state. The second curve, in top-down order from the figure, shows that random distribution of vaccination could not prevent the virus from entering the endemic state. However, distributing vaccination to individuals with big ranking in our pagerank-like model makes the epidemic die out quickly. This simple experiment confirms the important implication of our infection vector for the control of epidemic spread.

7. Acknowledgements

We would like thank Fabrcio Benevenuto from Federal University of Ouro Preto for the twitter graph, Kim Capps from Yahoo! Labs for his help to get access to Alta Vista web graph.

References

- [1] K. Bisset, Urgent computing for interaction based socio-technical simulations, invited presentation to Argonne National Laboratory (April).
- [2] K. Bisset, J. Chen, X. Feng, V. A. Kumar, M. Marathe, Epifast: A fast algorithm for large scale realistic epidemic simulations on distributed memory systems, in: Proceedings of 23rd ACM International Conference on Supercomputing (ICS'09), 2009, pp. 430–439.
- [3] A. Barabási, R. Albert, Statistical mechanics of complex networks, *Rev. Mod. Phys.* 74 (2002) 47–97.
- [4] M. Bowman, S. K. Debray, L. L. Peterson, Immunization of networks with community structure, *New Journal of Physics* 11 (123018).
- [5] M. E. J. Newman, The spread of epidemic disease on networks, *Phys. Rev. E* 66 (016128).
- [6] S. Bornholdt, T. Rohlf, Topological evolution of dynamical networks: Global criticality from local dynamics, *Phys. Rev. Lett.* 84 (2000) 6114–6117.
- [7] P. Holme, G. Ghoshal, Dynamics of networking agents competing for high centrality and low degree, *Phys. Rev. Lett.* 96 (098701).
- [8] T. Gross¹, B. Blasius, Adaptive coevolutionary networks: a review, *J. R. Soc. Interface* 5 (2008) 259–271.
- [9] P.-S. Romualdo, V. Alessandro, Epidemic spreading in scale-free networks, *Phys. Rev. Lett.* 86 (2001) 3200–3203. doi:10.1103/PhysRevLett.86.3200.
URL <http://link.aps.org/doi/10.1103/PhysRevLett.86.3200>
- [10] Modeling community containment for pandemic influenza: A letter report, Tech. rep. (2006).
- [11] P. Erdős, A. Rényi, On the evolution of random graphs, *Publ. Math. Inst. Hung. Acad. Sci.* (1960).
- [12] A. Barabási, R. Albert, Emergence of scaling in random networks, *Science* 286 (5439) (1999) 509–512.

- [13] M. E. J. Newman, The structure and function of complex networks, SIAM Rev. 45 (2003) 167–256.
- [14] M. E. J. Newman, Assortative mixing in networks, Phys. Rev. Lett. 89 (2007) 208701.
- [15] M. E. J. Newman, Spread of epidemic disease on networks, Phys. Rev. E. 66 (2002) 016128.
- [16] M. J. Keeling, K. T. Eames, Networks and epidemic models, J. R. Soc. Interface 2 (2005) 295–307.
- [17] M. E. J. Newman, Random graphs as models of networks, Wiley-VCH Verlag GmbH and Co. KGaA, 2005, pp. 35–68. doi:10.1002/3527602755.ch2. URL <http://dx.doi.org/10.1002/3527602755.ch2>
- [18] D. J. Watts, S. H. Strogatz, Collective dynamics of small-world networks, Nature 393 (1998) 440–442.
- [19] D. J. Watts, Small worlds: the dynamics of networks between order and randomness, Princeton University Press, Princeton, 1999.
- [20] M. E. J. Newman, D. J. Watts, Renormalization group analysis of the small-world network model, Phys. Lett. A. 263 (1999) 341–346.
- [21] C. Moore, M. E. J. Newman, Epidemics and percolation in small-world networks, Phys. Rev. E. 61 (2000) 5678–5682.
- [22] R. Cohen, S. Havlin, D. Ben-Avraham, Efficient immunization strategies for computer networks and populations, Phys. Rev. Lett. 91 (2003) 247901.
- [23] N. Madar, T. Kalisky, R. Cohen, D. Ben-Avraham, S. Havlin, Immunization and epidemic dynamics in complex networks, Eur. Phys. J. B. 38 (2) (2004) 269–276.
- [24] R. Albert, H. Jeong, A. Barabási, Error and attack tolerance of complex networks, Nature 406 (2000) 378–382.
- [25] R. Cohen, K. Erez, D. ben Avraham, S. Havlin, Resilience of the internet to random breakdowns, Phys. Rev. Lett. 85 (2000) 4626–4628. doi:10.1103/PhysRevLett.85.4626. URL <http://link.aps.org/doi/10.1103/PhysRevLett.85.4626>

- [26] Y. Wang, D. Chakrabarti, C. Wang, C. Faloutsos, Epidemic spreading in real networks: An eigenvalue viewpoint, in: In SRDS, 2003, pp. 25–34.
- [27] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: Bringing order to the web (1999).
- [28] K. Bryan, T. Leise, The \$25,000,000,000 eigenvector: The linear algebra behind google, SIAM Rev. 48 (3) (2006) 569–581. doi:10.1137/050623280.
URL <http://dx.doi.org/10.1137/050623280>
- [29] I. C. F. Ipsen, T. M. Selee, Pagerank computation, with special attention to dangling nodes, SIAM J. Matrix Anal. Appl. 29 (4) (2007) 1281–1296. doi:10.1137/060664331.
URL <http://dx.doi.org/10.1137/060664331>
- [30] N. Eiron, K. S. McCurley, J. A. Tomlin, Ranking the web frontier, in: Proceedings of the 13th international conference on World Wide Web, WWW '04, ACM, New York, NY, USA, 2004, pp. 309–318. doi:10.1145/988672.988714.
URL <http://doi.acm.org/10.1145/988672.988714>
- [31] D. Gleich, L. Zhukov, P. Berkhin, Fast parallel pagerank: a linear system approach, Tech. Rep. L-2004-038, Yahoo! Research Labs (2004).
- [32] T. H. Haveliwala, S. D. Kamvar, A. D. Kamvar, The second eigenvalue of the google matrix (2003).
- [33] J. W. Daniel, W. B. Gragg, L. Kaufman, G. W. Stewart, Reorthogonalization and stable algorithms for updating the gram-schmidt qr factorization, Math. Comp. 30 (1976) 772–795.
- [34] J. Demmel, J. Dongarra, A. Ruhe, H. van der Vorst, Templates for the solution of algebraic eigenvalue problems: a practical guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.
- [35] D. C. Sorensen, Implicitly restarted arnoldi/lanczos methods for large scale eigenvalue calculations, Tech. rep. (1996).
- [36] D. C. Sorensen, Implicit application of polynomial filters in a k-step arnoldi method, SIAM J. Matrix Anal. Appl. 13 (1) (1992) 357–385.

doi:10.1137/0613025.

URL <http://dx.doi.org/10.1137/0613025>

- [37] D. S. Watkins, The qr algorithm revisited (2008).
- [38] M. Bennani, T. Braconnier, Stopping criteria for eigensolvers, Tech. rep. (1994).